Dictionary of gray-level 3D patches for action recognition

Stefen Chan Wai Tim, Michèle Rombaut, Denis Pellerin

Dictionary of gray-level 3D patches for action recognition

Stefen Chan Wai Tim, Michele Rombaut, Denis Pellerin *

December 17, 2014

Abstract

This paper deals with action recognition in the context of video analysis based on the use of a sparse dictionary defined by the 3D spatio-temporal representation of the actions. A 3D volume can be seen as a set of gray-level 3D patches comprising 2D patches taken in successive frames in order to capture a motion pattern. The goal of our proposal is to recognize human actions within these 3D volumes whose 3D patches are described with the dictionary atoms. To that end, we compute a motion signature by building a histogram based on the use of the atoms of the dictionary. Paired with a SVM, we show that these signatures can be exploited in the context of action recognition. This method has been tested on the KTH database with good results.

1 Introduction

Action recognition research has developed a lot in the last years along with the rise of video contents, especially because its applications are numerous in surveillance, automatic video annotations or entertainment. Generally, it consists in extracting features either from each pixel of the whole image (or successive frames) [1], [2] or just from a few chosen pixels [3]. The goal is to classify some human activities using the data extracted from videos.

Action recognition often involves a lot of data and because of that, some techniques based on dictionaries and sparse representations [4], [5], [6] have emerged in the recent years to deal better with this aspect. These methods rely on the creation of a dictionary which can encode information contained in an image.

For the analysis of human activities, motion plays a crucial role. Usually, a motion descriptor can be computed (i.e Optical flow [7]) and the result is a 2D-based patch representation of the motion. Instead, we choose to analyse motion

*This work has been partially supported by the LabEx PERSYVAL-Lab (ANR-11-LABX-0025).

directly by using the spatio-temporal volume of gray-level pixels to extend 2D spatial patches without relying on an intermediary motion descriptor.

In this paper, we propose a method relying on a dictionary of 3D atoms (2 spatial dimensions and 1 temporal dimension) which can describe local motions to classify image sequences of a few frames extracted from a video containing human activities. The remainder of the paper is organized as follows: Section 2 presents the proposed action recognition method based on a dictionary of 3D atoms. Section 3 describes the experimental results. Finally, some conclusions are presented in Section 4.

2 Method framework

In this paper, we propose a method for human action recognition in the context of 2D video sequences. We want to classify spatio-temporal volumes composed of several frames and containing a human who perfoms a particular action. We assume that these volumes are previously defined. Some examples of such volumes can be observed on Figure 4.

The method described below builds a 3D patch-based dictionary from spatio-temporal volumes which can represent motion. We define a 3D image patch as a succession of 2D image patches (i.e the same patch and its variations through the successive frames of a video). Hence, the dictionary can be considered as a collection of patches called atoms (see Figure 1) which can be used to describe any patches as a combination of these atoms.

The presented algorithm can be decomposed into 3 main steps : (*i*) Dictionary learning, (*ii*) Signature computation and (*iii*) Classifier training.

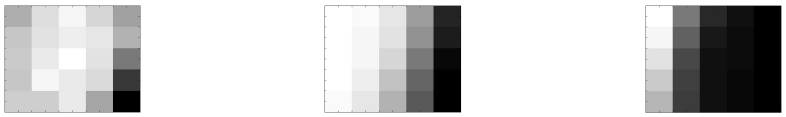
2.1 Dictionary learning

We have a dataset of spatio-temporal volumes at our disposal from which we can extract m 3D patches. From this set of 3D patches, we try to define the most representative dictionary. Different methods exist in the literature to learn dictionaries for sparse representation [8], [9]. We tested different dictionary learning algorithms available in the state of the art [10] and we did not notice any major difference in the results obtained by our method. Therefore, in our proposal, we decided to use the well-known K-SVD algorithm [9] to carry out the learning of the dictionary.

Starting with the notations, let $\mathbf{p}$ be a 3D patch of size $(s \times s \times t)$, with s being the size of a square patch in the spatial dimensions and t being the number of frames considered in the temporal dimension. This patch $\mathbf{p}$ is reshaped and treated as a column vector of size $n = s^2t : \mathbf{p} = (a_i)_{i\in[1,n]}$ with a_i a feature (gray level) associated to the pixel i. Each patch is normalized as : $\mathbf{p}_{norm} = \frac{\mathbf{p}-\bar{\mathbf{p}}}{\|\mathbf{p}\|^2}$ where $\bar{\mathbf{p}}$ is the mean of the vector $\mathbf{p}$ and $\| \mathbf{p} \|^2$ is the L_2-norm.

Let $Y = [\mathbf{p}^1_{norm}, \mathbf{p}^2_{norm}, ..., \mathbf{p}^m_{norm}] \in \mathbf{R}^{n\times m}$ be a a matrix composed of 3D patches, with m being the size of the dataset and $D = [\mathbf{d}_1, \mathbf{d}_2, ..., \mathbf{d}_{N_{atoms}}] \in$

$\mathbf{R}^{n \times N_{atoms}}$ be the dictionary of N_{atoms} atoms $\mathbf{d}_k$, where the number N_{atoms} is chosen empirically.



$t = 1$ $t = 2$ $t = 3$

Figure 1: Example of a 3D atom of $(s \times s \times t) = (5 \times 5 \times 3)$ pixels from a dictionary trained with the KTH database. This atom represents a horizontal motion going from right to left.

The formulation of the dictionary learning algorithm is :

$$\min_{D,X}\{\|Y - DX\|_F^2\} \text{ such that } \forall i \in [1, m], \|\mathbf{x}_i\|_0 \leq T_0 \tag{1}$$

where $X = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m] \in \mathbf{R}^{N_{atoms} \times m}$ contains the coefficients of the decomposition of Y using the dictionary D. $\mathbf{x}_i = (\alpha_j)_{j \in \|1, N_{atoms}\|}$ is a column vector from X and $\|\mathbf{x}_i\|_0$ is the norm that counts the number of non-zero entry of the vector. $\|\cdot\|_F$ is the Frobenius norm: $A \in \mathbf{R}^{n \times m}$, $\|A\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} |a_{ij}|^2}$. T_0 is the maximum of non-zero entries.

K-SVD is an iterative algorithm performed in two steps on the dataset Y. In the first step, the codes X are optimized with respect to a fixed dictionary D (Sparse coding):

$$X^{(k+1)} = \arg\min_{X}\{\|Y - D^{(k)}X\|_F^2\} \tag{2}$$

such that $\forall i \in [1, m], \|\mathbf{x}_i\|_0 \leq T_0$.
This step can be done using the Orthogonal Matching Pursuit (OMP) [11] algorithm which is a greedy algorithm used for its efficiency.

The second step consists in the update of the dictionary D with respect to X and is done according to (Dictionary update):

$$D^{(k+1)} = \arg\min_{D}\{\|Y - DX^{(k+1)}\|_F^2\} \tag{3}$$

The initial dictionary $D^{(0)}$ is initialized randomly with patches from the learning set.

The output is an overcomplete dictionary D composed of N_{atoms} whose atoms can be seen as elementary motion patterns as shown in Figure 2.

2.2 Signature computation

This section explains how to compute the signature of a spatio-temporal volume containing a moving object in a video sequence. Our implementation requires

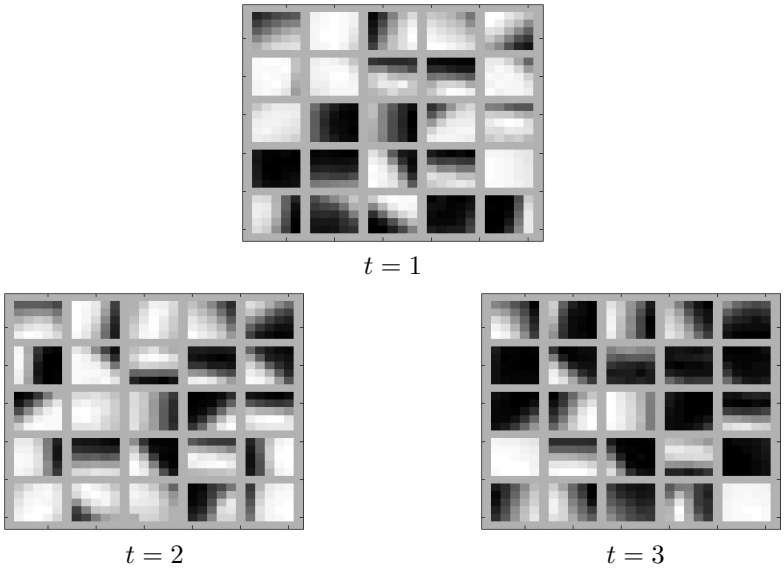


Figure 2: Example of 25 atoms of size $(s \times s \times t) = (5 \times 5 \times 3)$ pixels from a learned dictionary of KTH database. The 3 images taken together can describe elementary motion patterns.

a pre-processing step which consists in finding the localization of the humans to define the spatio-temporal volumes. This step is necessary to narrow down the region where motion occurs and can be done by using a pedestrian detector such as [12]. A spatio-temporal volume V can be decomposed into a set of overlapping 3D patches (see Figure 3). When the dictionary D is built, every patch $\mathbf{p}_{norm}$ of the volume V can be described as a linear combination of atoms $\mathbf{d}_k$ (Equation 4).

$$\mathbf{p}_{norm} = \left(\sum_{k=1}^{N_{atoms}} \alpha_k \cdot \mathbf{d}_k \right) + \mathbf{e}_{err} \tag{4}$$

where α_k is the coefficient of atom $\mathbf{d}_k$ and $\mathbf{e}_{err}$ is the reconstruction error.

The signature is defined as the histogram $\mathbf{h}_V$ of the volume V characterizing the motion inside the considered spatio-temporal volume. It is computed over this spatio-temporal volume by summing the sparse coefficients α_k of each atom of the dictionary D for all the patches of the volume.

Formally, each bin h_V^k associated to the atom $\mathbf{d}_k$ of the histogram is computed according to :

$$\forall k \in [1, N_{atoms}], h_V^k = \sum_{p_{norm} \in V} |\alpha_k| \tag{5}$$

with $\mathbf{h}_V = (h_V^k)_{k \in \|1, N_{atoms}\|}$.

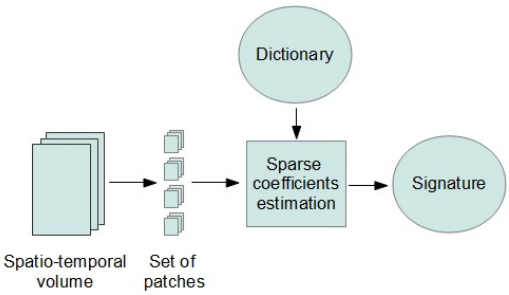


Figure 3: The spatio-temporal volume is decomposed into an overlapping set of patches $\mathbf{p}$ which are also decomposed into a set of coefficients α_k using the learned dictionary D. These coefficients are used to build the signature which is the histogram $\mathbf{h}_V$.

In order to handle the use of regions of different sizes, the histogram is then normalized, such that :

$$\sum_{k=1}^{N_{atoms}} h_V^k = 1 \tag{6}$$

As a consequence, a spatio-temporal volume can be represented using a histogram. An example of the resulting histogram is displayed in Figure 4.

2.3 Classifier training

The histogram (signature) $\mathbf{h}_V$ described in section 2.2 can model the human motion inside the volume V which is a region of a video and can consequently be used to train a classifier in the context of action recognition using only a few frames.

For the training, signatures are computed for labeled spatio-temporal volumes. These signatures are then given as input to a classical multiclass SVM classifier (Figure 5). The SVM is trained using a RBF kernel.

2.4 Spatial and temporal arrangement

Since a histogram representation discards the spatial and temporal dependencies, decomposing the spatio-temporal volume into smaller ordered volumes can give better results [13], [14]. Moreover, we do not want to increase the number of parameters describing this spatio-temporal volume.

Therefore, in order to improve our representation, we divide the spatio-temporal volumes into spatio-temporal cells (Figure 6). For 3D patches, cells can be formed in both spatial and temporal dimensions. In the spatial dimensions, we can divide the volume into equally distributed smaller volumes which describe the motion at a specific location. In the temporal dimension, it can be done by using multiple blocks of frames: for example, we can use $n_t = 2$ volumes

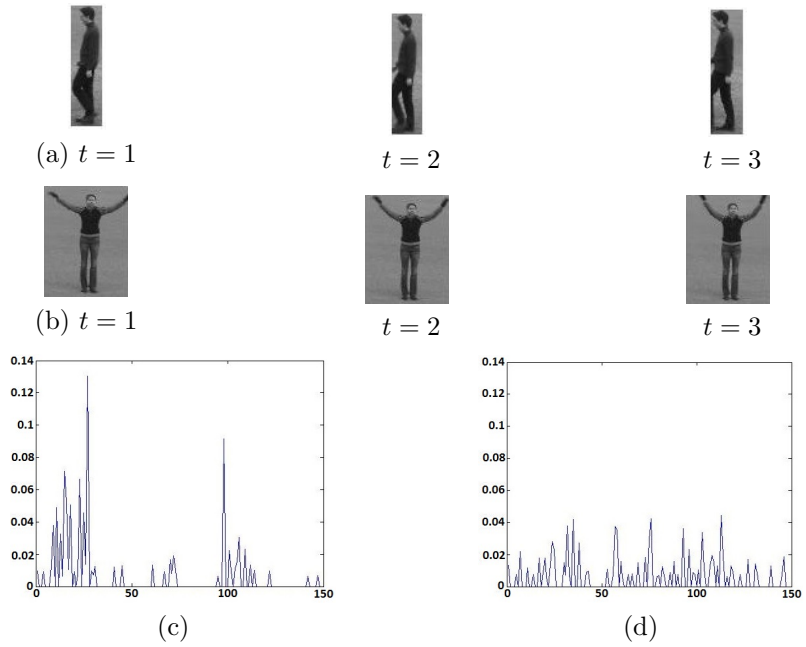


Figure 4: Two examples of spatio-temporal volumes containing human actions and the corresponding histogram (signature) computed with a dictionary of $N_{atoms} = 150$ trained with the KTH database: the spatio-temporal volume (a) gives the signature (c) of a walking action and the spatio-temporal volume (b) gives the signature (d) of a handwaving action.

by computing a signature for one volume made with t frames, coupled with the signature of a second volume with t frames which results in a signature of twice the size of the original signatures. The final signature used for classification is the concatenation of the feature histograms computed for each spatio-temporal cell. Experimentally, we found that using more cells yield an improved accuracy, while the dimension of the features rises accordingly.

Figure 5: A classic multiclass SVM is trained using labeled signatures as input.

(a) Spatial cells ($n_s = 4$)

(b) Temporal cells ($n_t = 2$)

Figure 6: Examples of "decomposition" of a spatio-temporal volume into (a) spatial and (b) temporal cells. Spatial cells are obtained by cutting the spatio-temporal volume into $n_s = 4$ parts. Temporal cells are obtained by juxtaposing $n_t = 2$ consecutive spatio-temporal volumes.

3 Experimentations

The proposed algorithm has been tested on the KTH database [15] which is a large public dataset containing videos of 25 people executing 6 types of actions (Walking, Running, Jogging, Boxing, Handwaving, Handclapping) in 4 different scenarios. These scenarios include indoor and outdoor sequences, changes in clothes and scale variations (Figure 7).

For the tests, we took 15 people divided in 2 groups, 10 for training / validation and 5 for testing. This is the usual ratio with this dataset [15], [16]. For each video, the spatio-temporal volumes corresponding to the human are manually annotated in a few frames, resulting in 10 computed signatures and a dataset of about 3600 samples. During the testing, classification labels are given for each sample.

Using this dataset, we aim to perform a classification of volumes using only a few frames. Regarding the parameters of the algorithm, we took as the reference the size $(s \times s \times t) = (5 \times 5 \times 3)$ for the 3D patches, $N_{atoms} = 150$ for the dictionary size and $T_0 = 1$ to define the number of atoms used to decompose one patch.

The dictionary is learned using the patches extracted from the spatio-temporal volumes of one person from the dataset for all actions and all scenarios. The SVM classifier is trained using a cross-validation paired with a gridsearch for the parameters. The results presented below are the results of the classification for the motion signature of each spatio-temporal volume taken independantly: no label is given to the videoclips.

The parameters that can be changed in the method are the patch size ($s \times s \times t$), the number of atoms in the dictionary N_{atoms}, the number of non-zero entry T_0 used for the decomposition of 3D patches and the number of temporal and spatial cells, n_t and n_s respectively, to take into account sequentiality and spaciality. We explored many changes in the parameters and refinements to

(a) Boxing

(b) Handclapping

(c) Handwaving

(d) Walking

(e) Jogging

(f) Running

Figure 7: Example of the actions performed within the KTH Dataset : "Boxing", "Handclapping", "Handwaving", "Walking", "Jogging" and "Running". Each action is executed in different conditions: indoor and outdoor, with clothes and scale variations.

assess the limits of the proposed signature. The results for some of the setups are given in Table 3.

First, for the choice of T_0 we found that $T_0 = 1$ gives out slightly better results than $T_0 = 5$ or $T_0 = 7$. Moreover, the number of atoms contained in the dictionary and the choice of the patch size also have an influence on the results. We tested different patch sizes: $t = 3$ and $t = 2$ for the size in the temporal dimension. The results are slightly better for $t = 3$. Another reason is to emphasize the spatio-temporal volume aspect. For the spatial dimensions, $s = 9$ and $s = 5$ lead to similar results. We evaluated 3 dictionary sizes: $N_{atoms} = 50$, $N_{atoms} = 150$ and $N_{atoms} = 250$. The classification performances decrease a little for the smaller dictionary and do not vary much between the two others. We found that $N_{atoms} = 150$ is a good choice for this dataset because the dimension of the signature increases rapidly when we use the division into cells.

We remind that the final signature is the concatenation of the signatures of the different cells (section 2.2). In the configuration, the spatio-temporal volume is divided into $n_s = 4$ regions in the spatial domain and into $n_t = 2$ regions into the temporal domain resulting in 8 spatio-temporal cells in totals. As a consequence, the final signature size is 1200. The choice of using temporal and spatial cells greatly improved our result from 68.25% to 81.67%.

The confusion matrix given in Table 3 shows the repartition of classification errors for the best configuration.

In [16], some classification results are given for the same dataset using various spatio-temporal features. We are in totally different conditions in our experi-

N_{atoms}	T_0	Patch size	n_t	n_s	Acc.
150	5	$5 \times 5 \times 3$	1	1	60.61%
150	1	$5 \times 5 \times 3$	1	1	68.25%
150	1	$9 \times 9 \times 3$	1	1	70.00%
50	1	$5 \times 5 \times 3$	1	1	65.92%
250	1	$5 \times 5 \times 3$	1	1	67.83%
150	1	$5 \times 5 \times 3$	2	1	66.75%
150	1	$5 \times 5 \times 3$	1	4	80.17%
150	1	$9 \times 9 \times 3$	1	4	79.25%
150	1	$5 \times 5 \times 3$	2	4	81.67%
150	1	$9 \times 9 \times 3$	2	4	75.33%

Table 1: Table containing the results of accuracy with different configurations of the dictionary or the feature. As we can see, adding $n_t = 2$ temporal and $n_s = 4$ spatial cells increases the classification accuracy since it gives back some spatial localization information which is lost when building a histogram.

	$Walk$	Box	$Wave$	$Clap$	Jog	Run
$Walk$	0.92	0	0	0	0.10	0.01
Box	0.01	0.84	0.04	0	0	0
$Wave$	0.02	0.15	0.77	0.05	0	0
$Clap$	0	0.05	0.19	0.95	0	0
Jog	0.05	0	0	0	0.75	0.30
Run	0	0	0	0	0.15	0.69

Table 2: Confusion matrix for a dictionary learned on $(5 \times 5 \times 3)$ patches, with 150 atoms and 8 spatio-temporal cells. Average accuracy for spatio-temporal volume classification is : 81.67%.

ment since we only evaluated a subset of the KTH database and only a small part of each video. However, in order to propose a comparison with these methods, we used the results of the volume classification as votes to label the videos. The class label assigned to a video corresponds to the class which contains most its volumes. In order to decide between 2 classes with equal number of votes, the longest succession of volumes assigned to the same class is taken into account. Table 3 shows the results obtained. As we can see the accuracy in this configuration increases: 85% instead of 81.67%. This result is due to the fact that even if we have some false classifications within a given video, the majority of frames are well classified resulting in a valid classification for the whole video.

Methods	Accuracy
HOG + Hessian	77.7%
HOG + Harris3D (detector)	80.9%
HOG/HOF + dense	86.1%
HOG/HOF + Harris3D (detector)	91.8%
Proposed method	85.0%

Table 3: Performance comparison for different features for the classification of videoclips. The results for the other features are obtained in [16] using histogram representations and Bag of Words model with a codebook size of 4000. For the proposed method, video labels were given by using the label given to the volumes as votes.

4 Conclusion

We have presented in this paper a new dictionary of 3D patches used to describe human motions. This new system can decompose spatio-temporal volumes into a motion signature that can be applied in the context of action recognition. One advantage of this method is that the classification of an action can be performed using only a reduced number of frames. This method has been tested on the KTH database with good results. Please note that the parameter choices presented in this paper depend on the database.

For future work, we are looking for dictionary reduction methods, to increase the dictionary effectiveness and build more complex signatures without increasing drastically their dimensions.

References

[1] A. A. Efros, A. C. Berg, G. Mori, and J. Malik, "Recognizing action at a distance," *ICCV*, 2003.

[2] K. Schindler and L. V. Gool, "Action snippets: How many frames does human action recognition require?," *CVPR*, 2008.

[3] P. Scovanner, S. Ali, and M. Shah, "A 3-dimensional sift descriptor and its application to action recognition," *Proceedings of the 15th international conference on Multimedia*, 2007.

[4] J. Yang, K. Yu, Y. Gong, and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification," *CVPR*, 2009.

[5] T. Guha and R. K. Ward, "Learning sparse representations for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, 2012.

[6] G. Somasundaram, A. Cherian, V. Morellas, and N. Papanikolopoulos, "Action recognition using global spatio-temporal features derived from sparse representations," *CVIU*, 2013.

[7] K. Jia, X. Wang, and X. Tang, "Optical flow estimation using learned sparse model," *ICCV*, 2011.

[8] T. Dean, G. Corrado, and R. Washington, "Recursive sparse, spatiotemporal coding," *Proceedings of the Fifth IEEE International Workshop on Multimedia Information Processing and Retrieval*, 2009.

[9] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, 2006.

[10] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Foundations and Trends in Machine Learning*, vol. 4, pp. 1–106.

[11] R. Rubinstein, M. Zibulevsky, and M. Elad, "Effcient implementation of the k-svd algorithm using batch orthogonal matching pursuit," *CS Technion*, 2008.

[12] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *PAMI*, 2010.

[13] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," *CVPR*, 2008.

[14] M. Marszalek, C. Schmid, H. Harzallah, and J. van de Weijer, "Learning object representations for visual object class recognition," *The PASCAL VOC07 Challenge Workshop, in conjonction with ICCV*, 2007.

[15] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach," *ICPR*, 2004.

[16] H. Wang, M. M. Ullah, A. Klaser, I. Laptev, and C. Schmid, "Evaluation of local spatio-temporal features for action recognition," *British Machine Vision Conference*, 2009.